

CLAIMS

What is claimed is:

- 1 1. A method for reducing lexical ambiguity in an input stream, comprising:
 - 2 breaking the input stream into at least two tokens;
 - 3 creating a connection graph using the at least two tokens, the connection graph
 - 4 comprising a plurality of paths;
 - 5 assigning a cost to each of the plurality of paths;
 - 6 defining at least one best path based upon a corresponding cost to generate an output
 - 7 graph; and
 - 8 providing the output graph to a syntactic analysis module to reduce lexical ambiguity.
- 1 2. The method of claim 1 wherein a number of the at least one best path is either
- 2 predefined or determined programmatically.
- 1 3. The method of claim 1 wherein creating a connection graph using the at least two
- 2 tokens comprises:
 - 3 compiling lexical grammar rules to generate lexical functions, the lexical grammar
 - 4 rules being written in a grammar programming language;
 - 5 creating a plurality of segments from the at least two tokens based upon lexical
 - 6 information and the lexical functions, and
 - 7 defining the plurality of paths using the plurality of segments.

1 4. The method of claim 1 wherein creating a connection graph using the at least two
2 tokens comprises assigning at least one part of speech tag to at least one of the at least two
3 tokens using lexical information.

1 5. The method of claim 1 wherein creating a connection graph using the at least two
2 tokens comprises recognizing a multiword expression in the input stream using multiword
3 information.

1 6. The method of claim 1 wherein the connection graph comprises a set of nodes and a
2 set of arcs.

1 7. The method of claim 6 wherein each of the plurality of paths comprises a combination
2 of nodes and arcs.

1 8. The method of claim 1 wherein the cost comprises lexical cost, unigram cost, bigram
2 cost and connector cost.

1 9. A method for providing segmentation of an input stream having at least two tokens,
2 comprising:

1 creating a plurality of segments from the at least two tokens based upon lexical
2 information and lexical functions; and

3 generating a connection graph using the plurality of segments.

1 10. The method of claim 9 further comprising compiling lexical grammar rules to generate
2 the lexical functions, the lexical grammar rules being written in a grammar programming
3 language.

1 11. The method of claim 10 wherein the lexical grammar rules define connectivity relation
2 of tokens.

1 12. The method of claim 9 further comprising assigning at least one part of speech tag to
2 at least one segment using a lexical dictionary.

1 13. The method of claim 12 further comprising:
2 defining a plurality of paths in the connection graph based upon part of speech tags
3 and the segments;
4 assigning a cost to each of the plurality of paths; and
5 determining at least one best path based upon a corresponding cost to generate an
6 output graph.

1 14. An apparatus for reducing lexical ambiguity in an input stream, comprising:
2 means for breaking the input stream into at least two tokens;
3 means for creating a connection graph using the at least two tokens, the connection
4 graph comprising a plurality of paths;
5 means for assigning a cost to each of the plurality of paths;
6 means for defining at least one best path based upon a corresponding cost to generate
7 an output graph; and

8 means for providing the output graph to a syntactic analysis module to reduce lexical
9 ambiguity.

1 15. The apparatus of claim 14 wherein a number of the at least one best path is either
2 predefined or determined programmatically.

1 16. The apparatus of claim 14 further comprising:
2 means for compiling lexical grammar rules to generate lexical functions, the lexical
3 grammar rules being written in a grammar programming language;
4 means for creating a plurality of segments from the at least two tokens based upon
5 lexical information and the lexical functions, and
6 means for defining the plurality of paths using the plurality of segments.

1 17. The apparatus of claim 14 further comprising means for assigning at least one part of
2 speech tag to at least one of the at least two tokens using lexical information.

1 18. The apparatus of claim 14 further comprising means for recognizing a multiword
2 expression in the input stream using multiword information.

1 19. The apparatus of claim 14 wherein the connection graph comprises a set of nodes and
2 a set of arcs.

1 20. The apparatus claim 19 wherein each of the plurality of paths comprises a combination
2 of nodes and arcs.

- 1 21. The apparatus of claim 14 wherein the cost comprises lexical cost, unigram cost,
2 bigram cost and connector cost.
- 1 22. An apparatus for providing segmentation of an input stream having at least two tokens,
2 comprising:
- 1 means for creating a plurality of segments from the at least two tokens based upon
2 lexical information and lexical functions; and
3 means for generating a connection graph using the plurality of segments.
- 1 23. The apparatus of claim 22 further comprising means for compiling lexical grammar
2 rules to generate the lexical functions, the lexical grammar rules being written in a grammar
3 programming language.
- 1 24. The apparatus of claim 23 wherein the lexical grammar rules define connectivity
2 relation of tokens.
- 1 25. The apparatus of claim 22 further comprising means for assigning at least one part of
2 speech tag to at least one segment using a lexical dictionary.
- 1 26. The apparatus of claim 25 further comprising:
2 means for defining a plurality of paths in the connection graph based upon part of
3 speech tags and the segments;
4 means for assigning a cost to each of the plurality of paths; and

5 means for determining at least one best path based upon a corresponding cost to
6 generate an output graph.

1 27. An apparatus for reducing lexical ambiguity in an input stream, comprising:
2 a tokenizer for breaking the input stream into at least two tokens;
3 a token connector for creating a connection graph using the at least two tokens, the
4 connection graph comprising a plurality of paths;
5 a cost assignor for assigning a cost to each of the plurality of paths;
6 a path calculator for defining at least one best path based upon a corresponding cost to
7 generate an output graph; and
8 a graph provider for providing the output graph to a syntactic analysis module to
9 reduce lexical ambiguity.

1 28. The apparatus of claim 27 wherein a number of the at least one best path is either
2 predefined or determined programmatically.

1 29. The apparatus of claim 27 wherein the token connector comprises:
2 a grammar programming language (GPL) compiler for compiling lexical grammar
3 rules to generate lexical functions, the lexical grammar rules being written in a general
4 programming language;
5 a segmentation engine for creating a plurality of segments from the at least two tokens
6 based upon lexical information and the lexical functions, and
7 a path designator for defining the plurality of paths using the plurality of segments.

1 30. The apparatus of claim 27 wherein the token connector comprises a part of speech
2 tagger for assigning at least one part of speech tag to at least one of the at least two tokens
3 using lexical information.

1 31. The apparatus of claim 27 wherein the token connector comprises a multiword
2 recognizer for recognizing a multiword expression in the input stream using multiword
3 information.

1 32. The apparatus of claim 27 wherein the connection graph comprises a set of nodes and
2 a set of arcs.

1 33. The apparatus of claim 32 wherein each of the plurality of paths comprises a
2 combination of nodes and arcs.

1 34. The apparatus of claim 27 wherein the cost comprises lexical cost, unigram cost,
2 bigram cost and connector cost.

1 35. An apparatus for providing segmentation of an input stream having at least two tokens,
2 comprising:
3 a segmentation engine for creating a plurality of segments from the at least two tokens
4 based upon lexical information and lexical functions; and
5 a graph generator for generating a connection graph using the plurality of segments.

1 36. The apparatus of claim 35 further comprising a grammar programming language
2 (GPL) compiler for compiling lexical grammar rules to generate the lexical functions, the
3 lexical grammar rules being written in GPL.

1 37. The apparatus of claim 36 wherein the lexical grammar rules define connectivity
2 relation of tokens.

1 38. The apparatus of claim 35 further comprising a part of speech tagger for assigning at
2 least one part of speech tag to at least one segment using lexical information.

1 39. The apparatus of claim 38 further comprising:
2 a path designator for defining a plurality of paths in the connection graph based upon
3 part of speech tags and the segments;
4 a cost assignor for assigning a cost to each of the plurality of paths; and
5 a path calculator for determining at least one best path based upon a corresponding
6 cost to generate an output graph.

1 40. A system for reducing lexical ambiguity, comprising:
2 a processor;
3 an input coupled to the processor, the input capable of receiving an input stream, the
4 processor configured to break the input stream into at least two tokens, create a connection
5 graph comprising a plurality of paths using the at least two tokens, assign a cost to each of the
6 plurality of paths, and define at least one best path based upon a corresponding cost to
7 generate an output graph; and

8 an output coupled to the processor, the output capable of providing the output graph to
9 a syntactic analysis module to reduce lexical ambiguity.

1 41. A system for providing segmentation of an input stream, comprising:
2 a processor;
3 an input coupled to the processor, the input capable of receiving an input stream
4 having at least two tokens, the processor configured to create a plurality of segments from the
5 at least two tokens based upon lexical information and lexical functions, and generate a
6 connection graph using the plurality of segments; and
7 an output coupled to the processor, the output capable of providing segmentation of
8 the input stream.

1 42. A computer readable medium comprising instructions, which when executed on a
2 processor, perform method for reducing lexical ambiguity in an input stream, comprising:
3 breaking an input stream into at least two tokens;
4 creating a connection graph using the at least one token, the connection graph
5 comprising a plurality of paths;
6 assigning a cost to each of the plurality of paths;
7 defining at least one best path based upon a corresponding cost to generate an output
8 graph; and
9 providing the output graph to a syntactic analysis module to reduce lexical ambiguity.

1 43. The computer readable medium of claim 42 wherein creating a connection graph
2 further comprises providing segmentation of the input stream using lexical information and
3 lexical functions.

1 44. The computer readable medium of claim 42 wherein creating a connection graph
2 further comprises assigning at least one part of speech tag to at least one of the at least two
3 tokens using lexical information.

1 45. The computer readable medium of claim 42 wherein creating a connection graph
2 further comprises recognizing a multiword expression in the input stream using lexical
3 information.

1 46. The computer readable medium of claim 42 wherein a number of the at least one best
2 path is either predefined or determined programmatically.

1 47. A computer readable medium comprising instructions, which when executed on a
2 processor, perform method for providing segmentation of an input stream having at least two
3 tokens, comprising:

4 creating a plurality of segments from the at least two tokens based upon lexical
5 information and lexical functions; and

6 generating a connection graph using the plurality of segments.

1 48. The computer readable medium of claim 47 further comprising compiling the lexical
2 grammar rules to generate lexical functions, the lexical grammar rules being written in a
3 grammar programming language.

1 49. A memory for storing data for access by an application program being executed on a
2 data processing system, comprising:
3 a data structure stored in said memory, said data structure including information
4 resident in a file used by said application program and including:
5 a plurality of packet structures used for the transmission of data, wherein each packet
6 structure includes
7 a set of nodes,
8 a set of arcs connecting at least two of the set of nodes, and
9 a value data object for each of the set of arcs having a value that represents a
10 corresponding part of speech tag.